

In the Claims

1. (Currently Amended) A process for enabling multiple programmers to modify behavior of an object executing on a computer system concurrently, the process comprising:
identifying a first method and a second method to be performed on an object, wherein the object corresponds to an instantiation of a class;
developing the first method in a first application having a first subclass of the class, wherein a first application-specific object is an instantiation of the first subclass;
and
concurrently developing the second method in a second application having a second subclass of the class, wherein a second application-specific object is an instantiation of the second subclass.
2. (Original) The process of claim 1 further comprising:
invoking the first method, wherein the invoking the first method on the first application-specific object such that the object communicates as if the first method were performed on the object.
3. (Original) The process of claim 1 further comprising:
invoking the second method, wherein the invoking the second method on the second application-specific object such that the object communicates as if the second method were performed on the object.
4. (Original) The process of claim 1 further comprising:
modifying the first method, wherein the modifying does not affect the second method.
5. (Original) The process of claim 1 further comprising:
modifying the second method, wherein the modifying does not affect the first method.
6. (Currently Amended) A process for enabling multiple programmers to modify behavior of an object executing on a computer system concurrently, the process comprising:
defining an abstract class for an object, the abstract class comprising:

a first method calling a first application; and
a second method calling a second application;
developing the first method in a first subclass of the abstract class in the first application;
and
developing the second method in a second subclass of the abstract class in the second application.

7. (Currently Amended) ~~An architecture~~ A system for enabling multiple programmers to modify behavior of an object executing on a computer system concurrently, the system comprising:

an object corresponding to an instantiation of a class;
a first application having a first subclass of the class, wherein
a first application-specific object is an instantiation of the first subclass;
the first subclass comprises a first method comprising a first behavior of the first application-specific object; and
the first behavior of the first application-specific object corresponds to a first behavior of the object;
a second application having a second subclass of the class, wherein
a second application-specific object is an instantiation of the second subclass;
the second subclass comprises a second method comprising a second behavior of the second application-specific object; and
the second behavior of the second application-specific object corresponds to a second behavior of the object.

8. (Currently Amended) ~~The architecture~~ system of claim 7 wherein invoking the first method performs the first method on the first application-specific object such that the object communicates as if the first method were performed on the object.

9. (Currently Amended) The ~~architecture~~ system of claim 7 wherein invoking the second method performs the ~~first~~ second method on the second application-specific object such that the object communicates as if the second method were performed on the object.
10. (Currently Amended) The ~~architecture~~ system of claim 7 wherein modifying the first method does not affect the second method.
11. (Currently Amended) The ~~architecture~~ system of claim 7 wherein modifying the second method does not affect the first method.
12. (Original) A computer program product comprising:
programming environment instructions for providing a programming environment comprising:
identifying instructions to identify a first method and a second method to be performed on an object; wherein
the object corresponds to an instantiation of class;
developing instructions to develop the first method in a first application having a first subclass of the class wherein a first application-specific object is an instantiation of the first subclass;
concurrent developing instructions to concurrently develop the second method in a second application having a second subclass of the class, wherein a second application-specific object is an instantiation of the second subclass;
and
a computer-readable medium to store the programming environment instructions, the identifying instructions, the developing instructions, and the concurrent developing instructions.
13. (Currently Amended) The computer program product of claim ~~10~~ 12 wherein invoking the first method performs the first method on the first application-specific object such that the object communicates as if the first method were performed on the object.

14. (Currently Amended) The computer program product of claim ~~10~~ 12 wherein invoking the second method performs the ~~first~~ second method on the second application-specific object such that the object communicates as if the second method were performed on the object.
15. (Currently Amended) The computer program product of claim ~~10~~ 12 wherein modifying the first method does not affect the second method.
16. (Currently Amended) The computer program product of claim ~~10~~ 12 wherein modifying the second method does not affect the first method.